

Propositional Logic Normal Forms

Literals

Definition

A **literal** is an atom or the negation of an atom. In the former case the literal is **positive**, in the latter case it is **negative**.

Negation Normal Form (NNF)

Definition

A formula is in **negation normal form (NNF)** if negation (\neg) occurs only directly in front of atoms.

Example

In NNF: $\neg A \wedge \neg B$

Not in NNF: $\neg(A \vee B)$

Transformation into NNF

Any formula can be transformed into an equivalent formula in NNF by pushing \neg inwards. Apply the following equivalences from left to right as long as possible:

$$\neg\neg F \equiv F$$

$$\neg(F \wedge G) \equiv (\neg F \vee \neg G)$$

$$\neg(F \vee G) \equiv (\neg F \wedge \neg G)$$

Example

$$(\neg(A \wedge \neg B) \wedge C) \equiv ((\neg A \vee \neg\neg B) \wedge C) \equiv ((\neg A \vee B) \wedge C)$$

Warning: “ $F \equiv G \equiv H$ ” is merely an abbreviation for
“ $F \equiv G$ and $G \equiv H$ ”

Does this process always terminate? Is the result unique?

CNF and DNF

Definition

A formula F is in **conjunctive normal form (CNF)** if it is a conjunction of disjunctions of literals:

$$F = \left(\bigwedge_{i=1}^n \left(\bigvee_{j=1}^{m_i} L_{i,j} \right) \right),$$

where $L_{i,j} \in \{A_1, A_2, \dots\} \cup \{\neg A_1, \neg A_2, \dots\}$

Definition

A formula F is in **disjunctive normal form (DNF)** if it is a disjunction of conjunctions of literals:

$$F = \left(\bigvee_{i=1}^n \left(\bigwedge_{j=1}^{m_i} L_{i,j} \right) \right),$$

where $L_{i,j} \in \{A_1, A_2, \dots\} \cup \{\neg A_1, \neg A_2, \dots\}$

Transformation into CNF and DNF

Any formula can be transformed into an equivalent formula in CNF or DNF in two steps:

1. Transform the initial formula into its NNF
2. Transform the NNF into CNF or DNF:
 - ▶ Transformation into CNF. Apply the following equivalences from left to right as long as possible:

$$\begin{aligned}(F \vee (G \wedge H)) &\equiv ((F \vee G) \wedge (F \vee H)) \\ ((F \wedge G) \vee H) &\equiv ((F \vee H) \wedge (G \vee H))\end{aligned}$$

- ▶ Transformation into DNF. Apply the following equivalences from left to right as long as possible:

$$\begin{aligned}(F \wedge (G \vee H)) &\equiv ((F \wedge G) \vee (F \wedge H)) \\ ((F \vee G) \wedge H) &\equiv ((F \wedge H) \vee (G \wedge H))\end{aligned}$$

Termination

Why does the transformation into NNF and CNF terminate?

Challenge Question: Find a weight function $w :: \text{formula} \rightarrow \mathbb{N}$ such that $w(l.h.s.) > w(r.h.s.)$ for the equivalences

$$\begin{aligned}\neg\neg F &\equiv F \\ \neg(F \wedge G) &\equiv (\neg F \vee \neg G) \\ \neg(F \vee G) &\equiv (\neg F \wedge \neg G) \\ (F \vee (G \wedge H)) &\equiv ((F \vee G) \wedge (F \vee H)) \\ ((F \wedge G) \vee H) &\equiv ((F \vee H) \wedge (G \vee H))\end{aligned}$$

Define w recursively:

$$w(A_i) = \dots$$

$$w(\neg F) = \dots w(F) \dots$$

$$w(F \wedge G) = \dots w(F) \dots w(G) \dots$$

$$w(F \vee G) = \dots w(F) \dots w(G) \dots$$

Complexity considerations

The CNF and DNF of a formula of size n can have size 2^n

Can we do better? Yes, if we do not insist on \equiv .

Definition

Two formulas F and G are **equisatisfiable** if F is satisfiable iff G is satisfiable.

Theorem

For every formula F of size n there is an equisatisfiable CNF formula G of size $O(n)$.